

Fig. 2 – Proposed BI-dashboard design

Conclusion. Developed dashboard is available in open GitHub repository [3], it could be easily deployed to free Power BI Desktop or Power BI SaaS (Software as a Service) platform, where further can be extended and/or customized by users.

References

1. Dong E., Du H., Gardner L. An interactive web-based dashboard to track COVID-19 in real time // The Lancet Infectious Diseases. – 2020.
2. JSON time-series of coronavirus cases // <https://github.com/pomber/covid19>
3. powerbi-covid-2019 // <https://github.com/andriikopp/powerbi-covid-2019>

ПІДТРИМКА ПРОЦЕСУ МОНІТОРИНГУ СТАНУ ОБЛАДНАННЯ ЗАСОБАМИ МАШИННОГО НАВЧАННЯ ТА TELEGRAM-БОТУ

Орловський Д.Л., Копп А.М., Литвинова В.С., Сизонова К.Г.

Національний технічний університет «Харківський політехнічний інститут»

Вступ і актуальність проблеми. Моніторинг стану обладнання, зокрема електричного, дозволяє знизити негативні наслідки його відмов, які зазвичай спричиняють порушення ізоляції, що можуть трапитися в будь-який час.

Згідно з сучасною концепцією індустрії 4.0, обробка даних, отриманих з IoT-сенсорів (internet of things, інтернет речей) пристроїв моніторингу, дозволяє сповіщувати працівників, відповідальних за профілактичне обслуговування, про стан працюючого обладнання [1].

Матеріали і методи. Розглянемо процес моніторингу стану обладнання, для якого наведено функціональну модель у нотації IDEF0 на рис. 1 [2]. Даний процес складається з етапів моніторингу умов роботи обладнання (A31), оцінки стану роботи системи (A32), сповіщення про відмови (A33), діагностики відмов (A34) та додаткових обчислень (A35). Входами процесу, що розглядається, є параметри роботи обладнання (I1, I2) та сигнали сповщення (I3, 32O2). До управляючих впливів належать методи моніторингу (C11) та оцінки стану обладнання (C12), а також метод аналізу сповіщень (C13). Виходами процесу є інформація щодо стану обладнання (31O, 32O1), сповіщення (33O), інформація про відмови (34O) та результати обчислень (35O). Пітримку виконання процесу забезпечує вимірювальне обладнання та програмне забезпечення (M1).

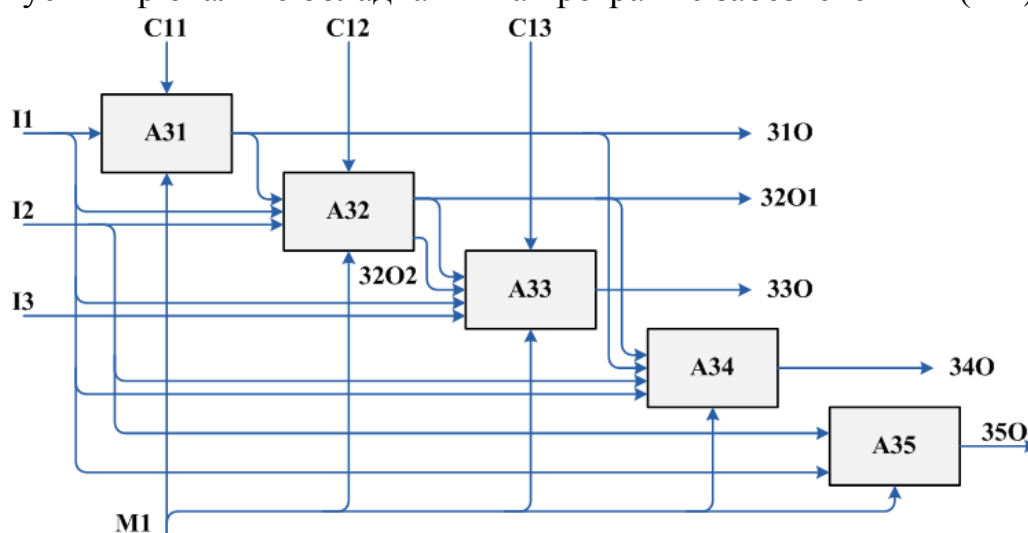


Рис. 1 – Модель процесу моніторингу стану обладнання

Недолік даного процесу полягає у відсутніх управляючих впливах блоків A34 та A35, що говорить про невизначеність щодо того, за яким регламентом виконується діагностика відмов та додаткові обчислення, пов'язані з аналізом результатів моніторингу обладнання. Оскільки модель відображає реальний процес моніторингу стану обладнання, наявні недоліки можуть мати негативний вплив на розглянуту діяльність. Для вдосконалення процесу потрібно обрати метод діагностики стану обладнання на основі базового показника – напруги. Дерево рішень, побудоване у RStudio, дозволяє з точністю 75.65% визначити компонент («comp1» або «comp2»), який вийшов з ладу (рис. 2). Дані, які були використані для побудови дерева рішень, доступні у системі Kaggle [3].

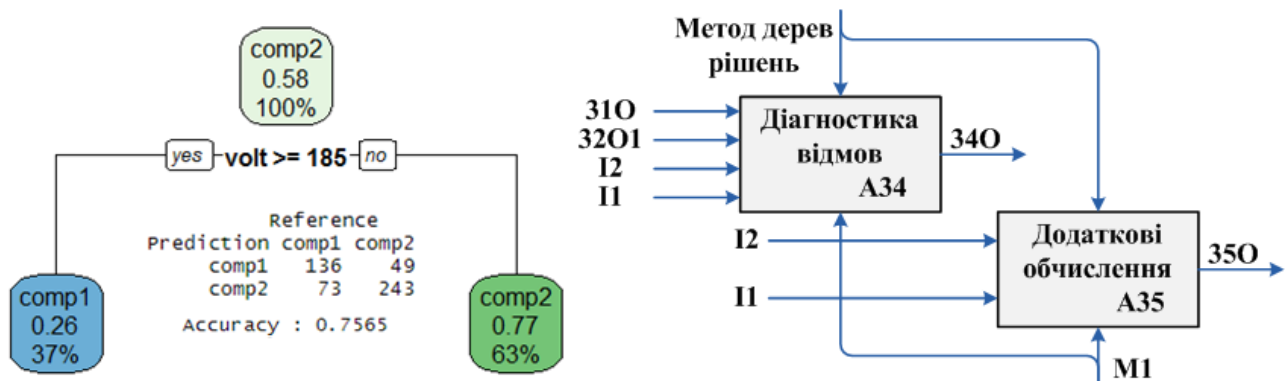


Рис. 2 – Діагностика відмов на основі методу дерев рішень

Результати. Впровадження сучасних та гнучких рішень потребує також і механізм сповіщення про відмови обладнання та інформацію щодо діагностики відмов. Тому пропонується прототип рішення на базі Telegram-бота, серверна частина якого буде отримувати дані з IoT-пристроїв, застосовувати метод дерев рішень для діагностики відмов та сповіщати про них у чат Telegram (рис. 3).

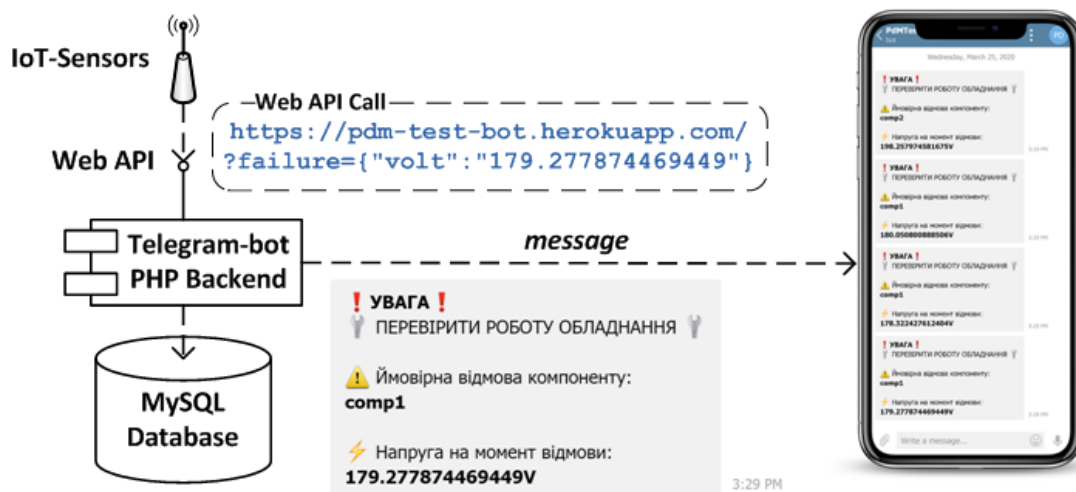


Рис. 3 – Архітектура запропонованого рішення

Висновки. Удосконалено процес моніторингу стану електрообладнання за рахунок використання дерев рішень для діагностики відмов та Telegram-боту для сповіщення про ймовірні відмови. Вихідний код доступний у GitHub [4], серверну частину рішення було розгорнуто у хмарній платформі Heroku.

Список використаних джерел

1. Heatley D., Abdel-Maguid M. The Internet of Things and Sustainable Manufacturing // Industry 4.0 and Engineering for a Sustainable Future. – Springer, Cham, 2019. – С. 91-115.
2. Chen Z. et al. Study on modeling of an integrated control and condition monitoring system for nuclear power plants // Progress of Nuclear Safety for Symbiosis and Sustainability. – Springer, Tokyo, 2014. – С. 57-68.
3. Predictive Maintenance // <https://www.kaggle.com/yuansaijie0604/xinjiang-pm/discussion>
4. andriikopp/pdm-test-bot // <https://github.com/andriikopp/pdm-test-bot>